

## Retraction Notice

Title of retracted article: **An Improved Fuzzy Based Algorithm for Detecting Text from Images Using Stroke Width Transform**

Author(s): Angamuthu Thilagavathy, Arulgnanaprakasam Chilambuchelvan  
 \* Corresponding author. Email: atv.cse@rmkec.ac.in

Journal: Circuits and Systems  
 Year: 2016  
 Volume: 7  
 Number: 4  
 Pages (from - to): 360 - 370  
 DOI (to PDF): <http://dx.doi.org/10.4236/cs.2016.74031>  
 Paper ID at SCIRP: 66062  
 Article page: <http://www.scirp.org/Journal/PaperInformation.aspx?PaperID=66062>  
 Retraction date: 2017-06-14

### Retraction initiative (multiple responses allowed; mark with X):

- All authors  
**X** Some of the authors:  
 Editor with hints from  Journal owner (publisher)  
 Institution:  
 Reader:  
 Other:

Date initiative is launched: 2017-06-09

### Retraction type (multiple responses allowed):

- Unreliable findings  
 Lab error  Inconsistent data  Analytical error  Biased interpretation  
 Other:  
 Irreproducible results  
 Failure to disclose a major competing interest likely to influence interpretations or recommendations  
 Unethical research
- Fraud  
 Data fabrication  Fake publication  Other:  
 Plagiarism  Self plagiarism  Overlap  Redundant publication \*  
 Copyright infringement  Other legal concern:
- Editorial reasons  
 Handling error  Unreliable review(s)  Decision error  Other:

**X** Other: The journal *Circuits and Systems* is not in the authors' university's list.

### Results of publication (only one response allowed):

- X** are still valid.  
 were found to be overall invalid.

### Author's conduct (only one response allowed):

- honest error  
 academic misconduct  
**X** none (not applicable in this case – e.g. in case of editorial reasons)

\* Also called duplicate or repetitive publication. Definition: "Publishing or attempting to publish substantially the same work more than once."

**History**

Expression of Concern:

yes, date: yyyy-mm-dd

no

Correction:

yes, date: yyyy-mm-dd

no

**Comment:**

This article has been retracted to straighten the academic record. In making this decision the Editorial Board follows [COPE's Retraction Guidelines](#). Aim is to promote the circulation of scientific research by offering an ideal research publication platform with due consideration of internationally accepted standards on publication ethics. The Editorial Board would like to extend its sincere apologies for any inconvenience this retraction may have caused.

# An Improved Fuzzy Based Algorithm for Detecting Text from Images Using Stroke Width Transform

Angamuthu Thilagavathy<sup>1\*</sup>, Arulgnanaprakasam Chilambuchelvan<sup>2</sup>

<sup>1</sup>Department of Computer Science and Engineering, RMK Engineering College, Kavaraipettai, India

<sup>2</sup>Department of Computer Science and Engineering, RMD Engineering College, Kavaraipettai, India

Email: <sup>\*</sup>atv.cse@rmkec.ac.in, chill94@gmail.com

Received 13 March 2016; accepted 25 April 2016; published 28 April 2016

Copyright © 2016 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## Abstract

The mounting escalation of multimedia content in information indexing and retrieval has made a high venture for extracting the text from videos. In this paper a new approach has been made using Stroke Width Transform (SWT) with intuitionistic fuzzy set theory for detecting text from images. The proposed methodology uses a fuzzy based edge detection method instead of the conventional edge detection methods. The fuzzy inference system is designed for edge detection. The edge detection is based on membership degree called hesitation degree and distance measure called intuitionistic fuzzy divergence. For text detection, SWT operator is used which tries to capture only the effectual text features. Geometric features of the text are used to differentiate between text and non-text regions of the text. The resulting system is tested on ICDAR 2003 dataset which produces promising results.

## Keywords

Edge Detection, Fuzzy Inference System, Intuitionistic Fuzzy Divergence, Stroke Width Transform

## 1. Introduction

Edge detection in image processing is an important task in the field of application of image processing. In this paper the two efficient techniques' image processing and soft computing are combined. Soft computing is a rising region for tribulations trading with uncertainties. Here fuzzy logic is used to detect the edges of images. Edge detection refers to the course of action of discovering pointed irregularities in an image. Soft computing is an up-and-coming promising field that includes fuzzy logic, genetic engineering, evolutionary computation and neural networks. The uncertainties may be due to noise, complex background, different orientations or intensi-

\*Corresponding author.

ties. In fuzzy set theory, every element is identified with a degree of membership. The degree of non-membership is equal to one minus the degree of membership. Usually for text detection using SWT [1], the conventional canny edge detection is used. In our system we use a fuzzy based edge detection method followed by SWT. In the proposed system, the image is segmented into  $3 \times 3$  binary matrices. Initially, we divide the image into divisions by means of a  $3 \times 3$  binary template matrix. Fuzzy inference system is intended to be with eight inputs that keep up a correspondence with eight pixels of the template matrix and one output. At the stage in fuzzification, the input image is transformed to the fuzzy domain [0 1]. Then the hesitation degree or intuitionistic fuzzy index is prognosticated. The utmost value of the divergence value among the sixteen fuzzy rule templates and the original image of the identical size is prognosticated. During defuzzification, the edge image is transformed back to the image pixel domain in the interval [1 255]. PSNR error metric is prognosticated to compare them with canny edge detection method and the proposed fuzzy edge detection method is found to be efficient. SWT is used to find the similarity between strokes based on their width. Initial value of each element in SWT is set to infinity. Letters in an image are provided with parallel sides and hence we find the edges of the texts using fuzzy. Next we calculate the gradient at each edges of the pixel. Letter candidates are then formed and filtering is carried out. The eligible pairs are then integrated together which give the detected text.

## 2. Related Work

The authors in [2] proposed a modified technique using fuzzy logic which is rule based. In [3], a fuzzy based technique was designed for detecting edge without setting the threshold value. In [4], the edge of the gray scale image is determined using fuzzy logic. This work is demonstrated along with the existing Sobel and prewitt edge detector. [5] proposed image edge detection based on soft computing approach which enhances the edge detection where the histogram is applied with fuzzy logic. [6] proposed edge detection using fuzzy logic in matlab. This work is based on sixteen rules, which differentiate the target pixel. [7] proposed an edge detection method without setting threshold value. It utilizes the nominal  $2 \times 2$  visor that skim over the entire image pixel by pixel. Fuzzy inference system and traditional edge operators are combined in [8]. In [9], the noise is removed at different level of processing. First and second derivative is implemented to FIS resultant image. In [10], heuristic rules were applied and a fuzzy based approach used to fulfill filtering and edge extraction simultaneously. Three linear spatial filters and spatial convolution techniques are used in [11]. A fuzzy based edge detection filter that passes through two stage of process to remove noise from grayscale images are used in [12]. SWT used in [13] to identify connected components and an unsupervised clustering to detect the text locations. The ICDAR dataset [14] [15] is the standard yardstick for text detection of natural images. In [16], a method which identifies texts regardless of its scale, direction, font and size are proposed. The method in [17] is based on corner points in a text. In [18], a Laplacian approach is used in order to detect text from video that handles text from any orientation. Later K-means clustering is achieved over the text. [19] employed a novel stroke like edge detection method and a temporal feature in extracting texts. Machine learning technique for scene text detection was used in [20]. They tested their system on ICDAR2005 and ICDAR2011 datasets. Maximally stable extremal regions (MSERs) are extracted using a pruning algorithm in [21]. A single-link clustering algorithm was proposed for grouping of text candidates. A text classifier was used to detect the text. In [22], a part based tree-structured models (TSMs) were proposed for the detection and recognition of text simultaneously. The Viterbi algorithm was used to improve the word recognition and the experimental results were conducted on ICDAR-2003 dataset. An unsupervised method was used in [23] to detect scene text where the text object was modelled as a pictorial structure. Three new character features was proposed. The performance was evaluated on ICDAR 2003/2005 dataset. In our earlier work [24], we proposed a hybrid approach by combining region and connected component (CC) based method where artificial neural network (ANN) was used as the classifier.

## 3. Methodology

An RGB image is taken as input. The original image is changed into gray scale image and converted in a fuzzy domain (0, 1) using fuzzy rules. Hesitation degree is prognosticated with the help of membership degree and non-membership degree for edge detection. Then, the divergence value is predicted and defuzzification method is implied. The stroke width at every pixel is extracted. Connected components are formed considering the stroke width of two neighboring pixels. Letter candidates are then grouped together which gives the detected text. The overall architecture of the proposed system is shown in **Figure 1**.

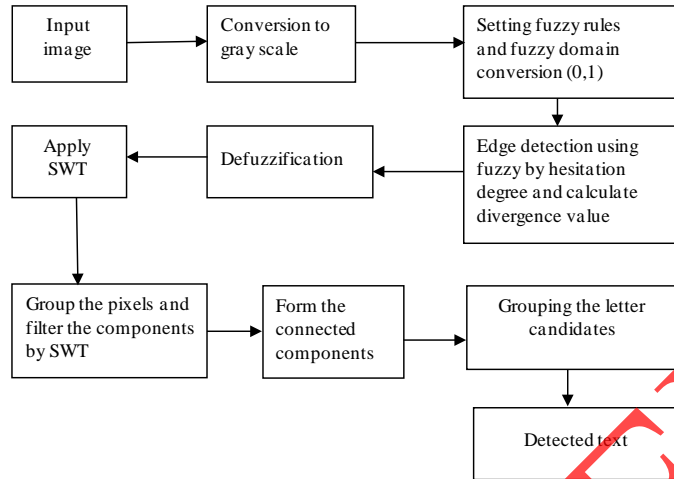


Figure 1. Overall workflow.

### 3.1. Intuitionistic Fuzzy Index

Fuzzy sets are sets with boundaries which are not accurate. The input pixels are separated into fuzzy sets explicitly black and white whereas the output pixel is separated into three fuzzy sets specifically black, white and edge. The membership in a fuzzy set is a matter of degree. If the fuzzy set is  $A$  and the relevant object is  $X$ , the proposition “ $X$  is a member of  $A$ ” may be true or false. A fuzzy set can be represented as:

$$A = \{ \langle X, \mu_A(X) \rangle; X \in E \} \tag{1}$$

where  $E$  is defined as the pervasive values, and  $\mu_A : E \rightarrow [0,1]$  where  $\mu_A$  defines the degree of membership. An intuitionistic fuzzy set (IFS)  $A$  is an entity of the outline:

$$A = \{ \mu_A(X), V_A(X); X \in E \} \tag{2}$$

where  $E$  is the pervasive set, and

$$\mu_A : E \rightarrow [0,1] \quad V_A : E \rightarrow [0,1] \tag{3}$$

The above equation defines the membership and non-membership degree respectively

$$0 < \mu_A(X) + V_A(X) < 1 \text{ and } \pi_A(X) = 1 - (\mu_A(X), V_A(X)) \tag{4}$$

The above equation is called the hesitation series.

### 3.2. Divergence Value

The maximum of the divergence value between the 16 fuzzy rule template and the original image of the same size is prognosticated. The fuzzy template is considered as a mask of  $3 \times 3$  and then it is slide over the fuzzy matrix. The divergence of  $3 \times 3$  matrixes is prognosticating with fuzzy rule template 1 to 16. Then the minimum element of the matrix for all 16 fuzzy rule templates is predicted. The divergence value is predicted as follows:

$$D_i = \left[ 2 - (1 - X_i) * \exp(X_i) - (1 + X_i) * \exp(-X_i) \right] + \left[ 2 - (1 - X_i + Y_i) * \exp(X_i - Y_i) (1 - Y_i + X_i) * \exp(Y_i - X_i) \right] \tag{5}$$

where  $X_i = A - t_i$ ,  $Y_i = p_i t_i - p_i A$  and  $m_i n_i = \min(\min(d_i))$ . Here  $D_i$  is the divergence with original image matrix and fuzzy rule template  $i$  where  $i$  varies from 1 to 16,  $A$  is the original image in  $3 \times 3$  matrix,  $t_i$  is the fuzzy template,  $p_i t_i$  is the intuitionistic fuzzy index.

In an image of size  $M \times M$  with  $L$  distinct gray levels having probabilities  $p_0, p_1, \dots, p_{L-1}$ , and the exponential entropy is defined as

$$H(A) = \frac{1}{n(\sqrt{e}-1)} \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} \left[ \frac{2\mu_A(a_{ij})\mu_B(b_{ij})}{\mu_A(a_{ij}) + \mu_B(b_{ij})} + \frac{2(1-\mu_A(a_{ij}))(1-\mu_B(b_{ij}))}{2-\mu_A(a_{ij})-\mu_B(b_{ij})} \right] \quad (6)$$

where  $n = M$  and  $i, j = 0, 1, 2, \dots, M-1$ , and  $\mu_A(a_{ij})$  is the membership degree of the  $(i,j)^{\text{th}}$  pixel  $a_{ij}$  in the image  $A$  and

$$\mu_A(a_{ij}) = \begin{cases} 0 & \text{if } \mu_A(a_{ij}) \leq 0.5 \\ 1 & \text{if } \mu_A(a_{ij}) > 0.5 \end{cases} \quad (7)$$

If  $A$  and  $B$  are two images then the membership degrees at  $(i,j)$  pixels for the images  $A$  and  $B$  is given by the following Equation (8). Based on entropy of fuzzy, the divergence between images  $A$  and  $B$  due to  $m_1(A)$  and  $m_1(B)$  may be given as

$$D_1(A, B) = \sum_{i=0}^n \sum_{j=0}^n \left[ \frac{\mu_A^2(a_{ij}) + \mu_B^2(b_{ij})}{\mu_A(a_{ij}) + \mu_B(b_{ij})} + \frac{(1-\mu_A(a_{ij}))^2 + (1-\mu_B(b_{ij}))^2}{2-\mu_A(a_{ij})-\mu_B(b_{ij})} \right] \quad (8)$$

Similarly, the divergence of  $B$  against  $A$  is:

$$D_1(B, A) = \sum_{i=0}^n \sum_{j=0}^n \left[ \sqrt{\frac{\mu_A^2(a_{ij}) + \mu_B^2(b_{ij})}{2}} + \sqrt{\frac{(1-\mu_A(a_{ij}))^2 + (1-\mu_B(b_{ij}))^2}{2}} \right] \quad (9)$$

So the total divergence between the pixels  $a_{ij}$  and  $b_{ij}$  of the images  $A$  and  $B$  due to  $m_1(A)$  and  $m_1(B)$  is given in Equation (10).

$$\begin{aligned} Div[m_1(A, B)] &= \sum_{i=0}^n \sum_{j=0}^n \left[ \frac{\mu_A^2(a_{ij}) + \mu_B^2(b_{ij})}{\mu_A(a_{ij}) + \mu_B(b_{ij})} + \frac{(1-\mu_A(a_{ij}))^2 + (1-\mu_B(b_{ij}))^2}{2-\mu_A(a_{ij})-\mu_B(b_{ij})} \right] \\ &+ \sum_{i=0}^n \sum_{j=0}^n \left[ \sqrt{\frac{\mu_A^2(a_{ij}) + \mu_B^2(b_{ij})}{2}} + \sqrt{\frac{(1-\mu_A(a_{ij}))^2 + (1-\mu_B(b_{ij}))^2}{2}} \right] \end{aligned} \quad (10)$$

Likewise, the total divergence between the pixels  $a_{ij}$  and  $b_{ij}$  of the images  $A$  and  $B$  due to  $m_2(A)$  and  $m_2(B)$  is

$$\begin{aligned} Div[m_2(A, B)] &= \sum_{i=0}^n \sum_{j=0}^n \left\{ \frac{\mu_A^2(a_{ij}) + \mu_B^2(b_{ij})}{\mu_A(a_{ij}) + \mu_B(b_{ij})} (\pi_B(b_{ij}) - \pi_A(a_{ij})) \right. \\ &+ \left. \frac{(1-\mu_A(a_{ij}))^2 + (1-\mu_B(b_{ij}))^2}{2-\mu_A(a_{ij})-\mu_B(b_{ij})} - \left[ \frac{\mu_A(a_{ij}) + \sqrt{\mu_A(a_{ij})\mu_B(b_{ij})} + \mu_B(b_{ij})}{3(\pi_B(b_{ij}) - \pi_A(a_{ij}))} \right] \right\} \end{aligned} \quad (11)$$

Thus the overall intuitionistic fuzzy divergence IFD between the images  $A$  and  $B$  is defined as

$$IFD(A, B) = m_1(A, B) + m_2(A, B) \quad (12)$$

The above Equation (12) gives the total divergence between the pixels  $a_{ij}$  and  $b_{ij}$  of the images  $A$  and  $B$  due to  $m_1(A)$  &  $m_1(B)$  and  $m_2(A)$  &  $m_2(B)$ . The text detection is based on the IFD.

### 3.3. Fuzzy Inference Rules

The inference rules are based on the weight of the eight neighborhood gray level pixels. These rules encompass the capability to haul out the edges in the developed gray image competently. The consequential image has the form of black and the white regions. The input which undergoes a process of fuzzification and the output which

follows the process of defuzzification pixel is ranged from 0 - 255. Finally the black, white and edge are detected. The following rules will decide the output pixels range of three decisions—white , black or edge fuzzy set. **Figure 2** represents the rules for fuzzy inference system. The values of  $a$  &  $b$  are chosen using trial and error method.

### 4. Text Detection

Stroke Width Transform SWT is used to calculate the width of the most likely stroke containing the pixel. The stroke-width variance is determined and should not be too large or too small. The variance of the stroke-width is set based on trial and error basis. The stroke width is the key parameter in forming the letter candidates. Connected component algorithm is used for grouping the text. The SWT Detector can detect letters of different languages like English, Hebrew, and Arabic etc. The text can be of varying sizes, different orientation. The SWT can also detect handwritten text. The algorithm depicting the overall process for detecting text is given below.

**Algorithm 1** Overview of the process followed for detecting text

```

Pre-process the input image I through conversion to gray scale G to obtain O
Generate fuzzy rules FR and fuzzy domain conversion on O
Detect the edge E on O based on FR by hesitation degree H and divergence value D
for i = 1 to ELEMENTS do
Calculate swt S on edge detected image E
Generate connected components CC after grouping and filtering the components by SWT
end for
    
```

### 5. Evaluation and Discussion

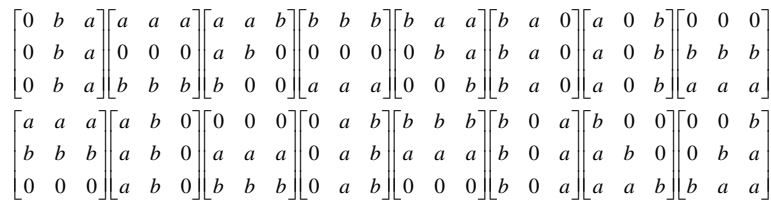
The ICDAR dataset [14] [15] is the standard yardstick for text detection of natural images. The ICDAR dataset consists of 258 images in the training set and 251 images in the test set. The images in the dataset are in full-color and it varies in size from  $307 \times 93$  to  $1280 \times 960$  pixels. We follow [1] for the evaluation of the proposed system. Our method is compared with respect to f-measure. The f-measure is in turn a combination of two other measures: precision and recall. Ground truth boxes called targets is made available in the dataset. The output which contains the bounding boxes (the green rectangles in **Figure 6**) for the correctly detected words are called estimate. The intersection area over the minimum bounding box area of both rectangles is called the match  $m_p$  between the two rectangles. The best match  $m(r; R)$  for a rectangle  $r$  among a set of rectangles  $R$  is defined as

$$m(r; R) = \max \{m_p(r; r_0) \mid r_0 \in R\} \tag{13}$$

The definitions for precision and recall are taken from [1].

$$\begin{aligned} \text{Precision} &= \frac{\sum r_e \in E^m(r_e, T)}{|E|} \\ \text{Recall} &= \frac{\sum r_i \in T^m(r_i, E)}{|T|} \end{aligned} \tag{14}$$

where  $T$  is the set of ground-truth and  $E$  is the set of estimated rectangles. Two more metric for performance evaluation is taken from [25] namely, Character-Recall (CR) and Word-Recall (WR). The CR is defined as the



**Figure 2.** Rules for fuzzy inference system.

ratio of number of characters detected to the total number of characters.

$$CR = \frac{\text{no.of characters detected}}{\text{total no.of characters}} \quad (15)$$

WR is defined as the ratio of the number of words detected to the total number of words.

$$WR = \frac{\text{no.of words detected}}{\text{total no.of words}} \quad (16)$$

In [25], five experiments were done and the CR ranges from 58.7% to 64.9% and WR ranges from 77.4% to 85.6%. In ICDAR 2003 competitions [15], the WR rate for the best performing system was 52%. In [1] Word Recall rate is 79.04%. For our system WR rate is 82% and CR is 63%. The result shows that fuzzy is more efficient than other edge detectors. For example: the canny operator cannot differentiate between real and unreal edges. So, it leads to low performance. But, in fuzzy edge detector it reduces noises in images and it improves blurred pixels to detect the edges of letters.

The smudge image pixel is improved by applying different threshold values to images. Therefore, fuzzy edge detector detects edges perfectly. **Figure 3** illustrates the comparison between canny edge detector and our proposed fuzzy edge detector. For the performance evaluation of our system, we first highlight the fuzzy detector output on the input image. Our fuzzy detector is evaluated based on the PSNR metric and is compared against the canny edge detector. We are comparing with canny edge detector because most of the algorithms [1] use canny edge detector along with SWT.

From **Figure 3** it can be observed that the obscure, blurred, slanting, inclined texts are veraciously succoured by the fuzzy edge detector. To assess the quality of the edge detected image PSNR value is taken as a metric to compare the quality of the image.

The PSNR is given by the following,

$$PSNR = 10 \log_{10} \left( \frac{R^2}{MSE} \right) \quad (17)$$

where,  $R$  is the pixel range,  $MSE$  is the mean square error.

The PSNR value is prognosticated for the original image across canny and proposed fuzzy edge detector. The PSNR block computes the peak signal-to-noise ratio between two images and is represented in decibels. The higher value of the PSNR, better the quality of the reconstructed image. From the PSNR calculation it is seen that fuzzy edge image has the higher PSNR value compared to canny edge detector. So this proves that the proposed Fuzzy edge detection method performs well. We performed PSNR calculation for almost 256 images which are procured from ICDAR 2003 dataset. In **Figure 4**, the higher peaks denotes better edge detection in fuzzy than canny edge detector. We then perform SWT technique on the image obtained from the fuzzy edge detector. Our system was able to detect text in many exigent scenarios such as non homogeneous background, shadowy & blurry images, slanting text, inclined text, non horizontal text, curved text, fonts of varying size. The examples shown in **Figure 5** are taken from the ICDAR 2003 dataset. We have less false negatives in **Figure 5**. **Figure 5** shows the results of our system on complex scenarios such as shadowy text, text overlapped on another text, text with small font size, non-horizontal text, and text with big font size. Green rectangles depict the text detected. **Figure 6** depicts the cases of fancy fonts where the green rectangles in (a)-(g) depict the text detected and (h)-(n) depicts the finally detected text based on our algorithm. **Figure 7** depicts the examples of complex scenarios which include shadowy text, text overlapped on another text as well as text with small font, non-horizontal text and text with big font.

**Figure 8** shows the results of failure cases. The failure for the text detection were due to too much blur in the text, text that is not clear, text with too small a font size and text with too much curve. We also tested our system on cute80 dataset [26]. The dataset contains 80 images exclusively of curved text. The example results of our system on cute80 dataset are shown in **Figure 9**. The performance of our system on cute80 dataset is as follows: Precision 67% and Recall 68%.

Performance comparison of text detection algorithms are provided in **Table 1** which gives the comparisons of precision, recall used by various algorithms which were tested on the ICDAR dataset. We have compared our system with other algorithms where the results are evaluated based on the ICDAR dataset.



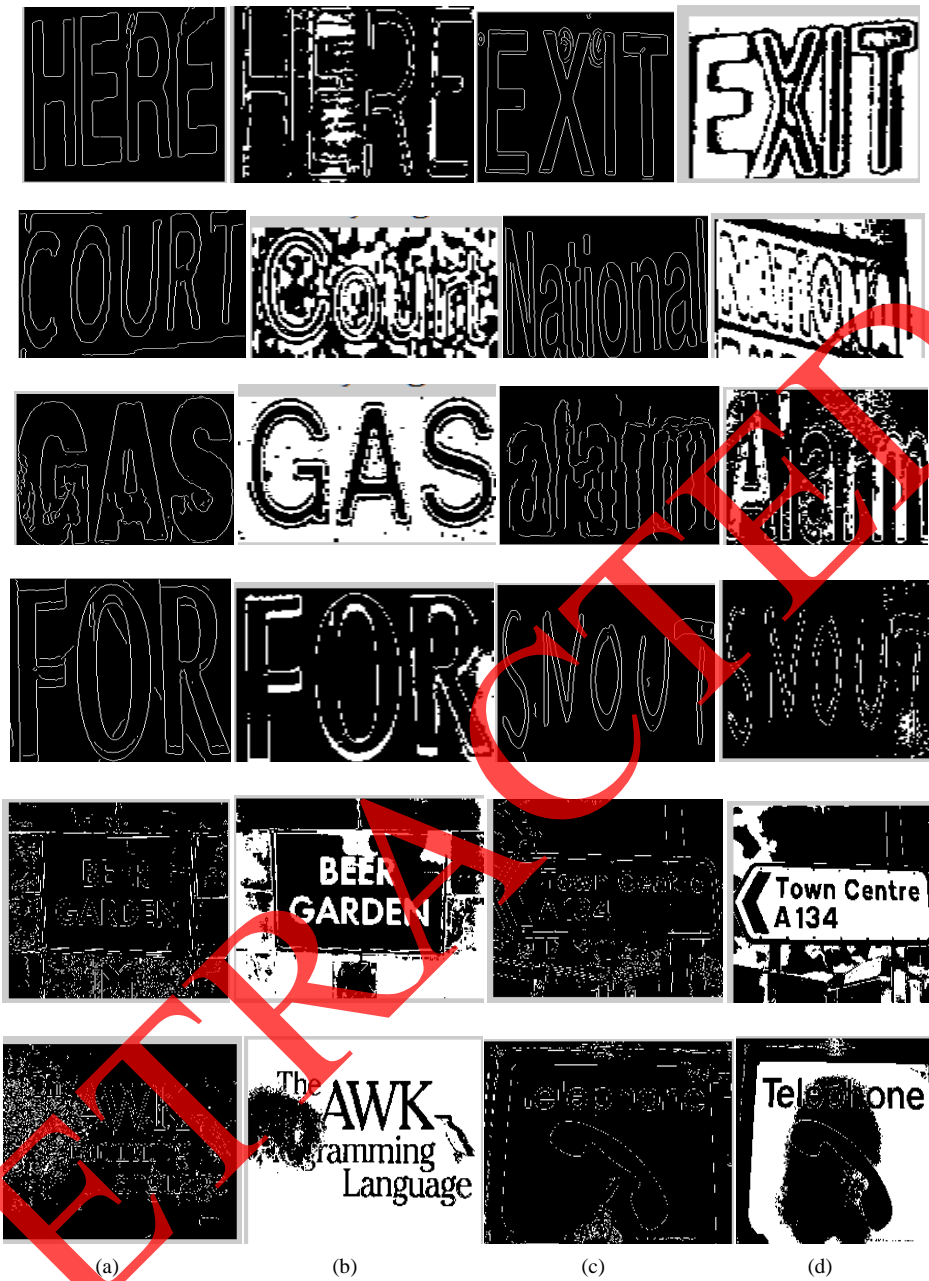


Figure 3. Comparison output of canny and fuzzy edge. (a) (c) Canny edge detection output; (b) (d) Fuzzy-based edge detection output.

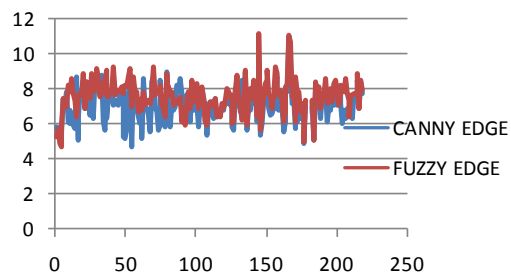


Figure 4. PSNR calculation.

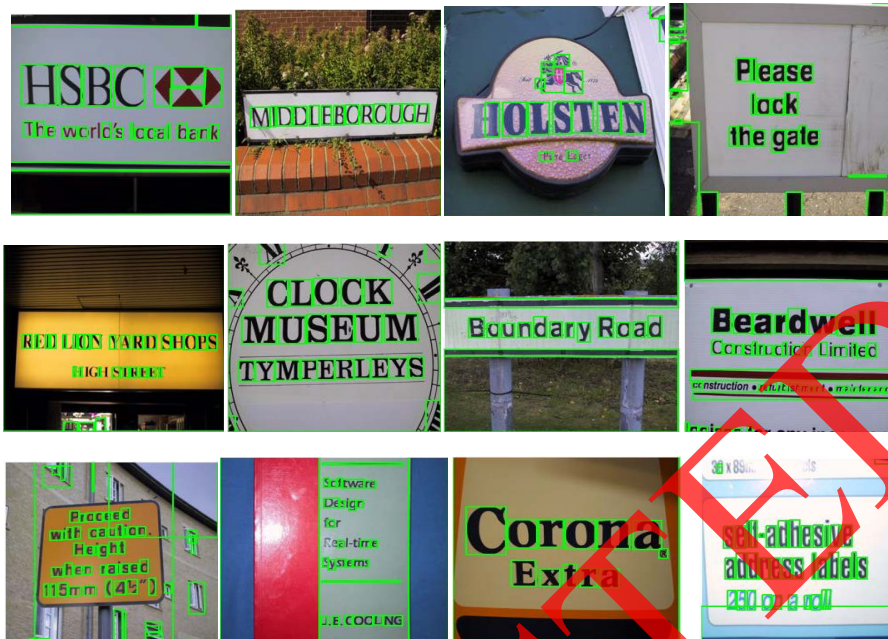


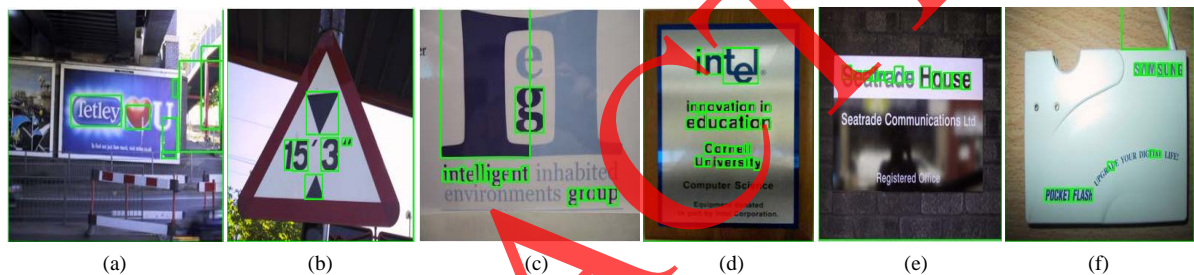
Figure 5. Examples of text detection results based on our algorithm. Green rectangles depict the text detected.



Figure 6. (a)-(g) Examples of fancy fonts detection. Green rectangles depict the text detected; (h)-(n) Finally detected text of our algorithm.



**Figure 7.** Examples of complex scenarios. (a) Shadowy text; (b) Text overlapped on another text as well as text with small font; (c) Non-horizontal text; (d) Text with big font. Green rectangles depict the text detected; (e)-(h) Finally detected text of our algorithm.



**Figure 8.** Examples of failure cases. (a) Too much blur text; (b) False positive; (c) Text that is not clear; (d) Too small font; (e) Transparent text; (f) Too much curved text.



**Figure 9.** Examples of curved text detection. Samples are taken from cute 80 dataset [26].

**Table 1.** Performance comparison of text detection algorithms.

Algorithm	Precision	Recall
Our system	0.81	0.72
Boris Epshtein [1]	0.73	0.60
Hinnerk Becker	0.62	0.67
Alex Chen	0.60	0.60
Qiang Zhu	0.33	0.40
Jisoo Kim	0.22	0.28
Nobuo Ezaki	0.18	0.36
Ashida	0.55	0.46
HWDavid	0.44	0.46
Wolf	0.30	0.44
Todoran	0.19	0.18
Full	0.1	0.06

## 6. Conclusion

In this paper we presented a new approach for text detection using fuzzy edge detection along with Stroke Width Transform (SWT). We first highlight the fuzzy detector output on the input image. It can be observed that the obscure, blurred, slanting, inclined texts are veraciously succoured by the fuzzy edge detector. Our fuzzy detector is evaluated based on the PSNR metric and is compared against the canny edge detector. We are comparing with canny edge detector because most of the algorithms use canny edge detector along with SWT. We then perform SWT technique on the image obtained from the fuzzy edge detector. We test our system on ICDAR 2003 dataset. Our system was able to detect text in many exigent scenarios such as non-homogeneous background, shadowy & blurry images, and text with fancy font, slanting text, inclined text, non-horizontal text, curved text, fonts of varying size. Our system is evaluated along with precision, recall with the following metrics: character recall and word recall. We also tested our system on cute80 dataset. The dataset contains 80 images exclusively of curved text. For our system WR rate is 82% and CR is 63% which demonstrate good potential in detecting text from images.

## References

- [1] Epshtein, B., Ofek, E. and Wexler, Y. (2010) Detecting Text in Natural Scenes with Stroke Width Transform. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, San Francisco, 13-18 June 2010, 2963-2970.
- [2] Kaur, M. and Kaur, S. (2011) A New Approach to Edge Detection Using Rule Based Fuzzy Logic. *Journal of Global Research in Computer Science*, **2**, 15-19. <http://dx.doi.org/10.7815/ijorcs.22.2012.017>
- [3] Alshennawy, A.A. and Aly, A. (2009) Edge Detection in Digital Images Using Fuzzy Logic Technique. *World Academy of Science, Engineering and Technology*, **3**, 540-548.
- [4] Ur Rahman Khan, A. and Thakur, K. (2012) An Efficient Fuzzy Logic Based Edge Detection Algorithm for Gray Scale Image. *International Journal of Emerging Technology and Advanced Engineering*, **2**, 245-250.
- [5] Khaire, P.A. and Thakur, N.V. (2012) Image Edge Detection Based on Soft Computing Approach. *International Journal of Computer Applications*, **1**, 12-14. <http://dx.doi.org/10.5120/8061-1426>
- [6] Suryakant, N.K. (2012) Edge Detection Using Fuzzy Logic in Matlab. *International Journal of Research in Computer Science and Software Engineering*, **2**, 38-40.
- [7] Bhagabati, B. and Das, C. (2012) Edge Detection of Digital Images Using Fuzzy Rule Based Technique. *International Journal of Advanced Research in Computer Science and Software Engineering*, **2**, 259-262.
- [8] Kaur, S.A.A. (2012) Modified Edge Detection Technique Using Fuzzy Inference System. *International Journal of Computer Applications*, **44**, 9-12.

- [9] Kaur, K., Mutenja, V. and Gill, I.S. (2010) Fuzzy Logic Based Image Edge Detection Algorithm in MATLAB. *International Journal of Computer*, **1**, 0975-8887. <http://dx.doi.org/10.5120/442-675>
- [10] Becerikli, Y., Karan, T.M. and Okatan, A. (2009) A New Fuzzy Edge Detector for Noisy Images Using Modified WFM Filter. *International Journal of Innovative Computing, Information and Control*, **5**, 1725-1733.
- [11] Aborisade, D.O. (2011) Novel Fuzzy Logic Based Edge Detection Technique. *International Journal of Advanced Science and Technology*, **29**, 75-821.
- [12] Ishta, A.G. and Gupta, R. (2014) Fuzzy Based New Algorithm for Noise Removal and Edge Detection. *International Journal of Advanced Research in Computer Science & Technology*, **2**, 191-196.
- [13] Mosleh, A. (2013) Automatic in Painting Scheme for Video Detection and Removal. *IEEE Transactions on Image Processing*, **22**, 4460-4472. <http://dx.doi.org/10.1109/TIP.2013.2273672>
- [14] Lucas, S.M. (2005) ICDAR 2005 Text Locating Competition Results. *Proceedings of 8th International Conference on Document Analysis and Recognition*, **1**, 80-84. <http://dx.doi.org/10.1109/ICDAR.2005.231>
- [15] Lucas, S.M., Panaretos, A., Sosa, L., Tang, A., Wong, S. and Young, R. (2003) ICDAR 2003 Robust Reading Competitions. *Proceedings of 7th International Conference on Document Analysis and Recognition*, Edinburgh, 6 August 2003, 682. <http://dx.doi.org/10.1109/ICDAR.2003.1227749>
- [16] Kumar, S. and Perrault, A. (2010) Text Detection on Nokia N900 using Stroke width Transform-Computer Vision.
- [17] Zhao, X., Lin, K.-H., Fu, Y., Hu, Y.X., Liu, Y.C. and Huang, T.S. (2011) Text from Corners: A Novel Approach to Detect Text and Caption in Videos. *IEEE Transactions on Image Processing*, **20**, 790-799. <http://dx.doi.org/10.1109/TIP.2010.2068553>
- [18] Shivakumara, P., Phan, T.Q. and Tan, C.L. (2011) A Laplacian Approach to Multi Oriented Text Detection in Video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **33**, 412-419. <http://dx.doi.org/10.1109/TPAMI.2010.166>
- [19] Liu, X.Q. and Wang, W.Q. (2012) Robustly Extracting Captions in Videos Based on Stroke-Like Edges and Spatio-Temporal Analysis. *IEEE Transactions on Multimedia*, **14**, 482-489. <http://dx.doi.org/10.1109/TMM.2011.2177646>
- [20] Il Koo, H. and Kim, D.H. (2013) Scene Text Detection via Connected Component Clustering and Nontext Filtering. *IEEE Transactions on Image Processing*, **22**, 2296-2305. <http://dx.doi.org/10.1109/TIP.2013.2249082>
- [21] Yin, X.-C., Yin, X.W., Huang, K.Z. and Hao, H.-W. (2014) Robust Text Detection in Natural Scene Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **36**, 970-983. <http://dx.doi.org/10.1109/TPAMI.2013.182>
- [22] Shi, C.-Z., Wang, C.-H., Xiao, B.-H., Gao, S. and Hu, J.-L. (2014) Scene Text Recognition Using Structure-Guided Character Detection and Linguistic Knowledge. *IEEE Transactions on Circuits and Systems for Video Technology*, **24**, 1235-1250. <http://dx.doi.org/10.1109/TCSVT.2014.2302522>
- [23] Zhang, J. and Kasturi, R. (2014) A Novel Text Detection System Based on Character and Link Energies. *IEEE Transactions on Image Processing*, **23**, 4187-4198. <http://dx.doi.org/10.1109/tip.2014.2341935>
- [24] Thilagavathy, A., Aarthi, K. and Chilambuchelvan, A. (2012) A Hybrid Approach to Extract Scene Text from Videos. *International Conference on Computing, Electronics and Electrical Technologies*, Kumaracoil, 21-22 March 2012, 1017-1022. <http://dx.doi.org/10.1109/icceet.2012.6203930>
- [25] Subramanian, K., Natarajan, P., Decerbo, M. and Castañón, D. (2005) Character-Stroke Detection for Text Localization and Extraction. *International Conference on Document Analysis and Recognition*, **1**, 30-37.
- [26] Risnumawan, A., Shivakumara, P., Chan, C.S. and Tan, C.L. (2014) A Robust Arbitrary Text Detection System for Natural Scene Images. *Expert Systems with Applications*, **41**, 8027-8048. <http://dx.doi.org/10.1016/j.eswa.2014.07.008>